

1A Array Stack (STL)

```
#include <bits/stdc++.h>
using namespace std;
int main(){stack<int> s;int c,x;while(cin>>c){
if(c==1)cin>>x,s.push(x);
else if(c==2){if(s.empty())cout<<"UF";else cout<<s.top(),s.pop();}
else if(c==3){auto t=s;while(!t.empty())cout<<t.top()<<" ",t.pop();}
else break;}}
```

1B Array Queue (STL)

```
#include <bits/stdc++.h>
using namespace std;
int main(){queue<int> q;int c,x;while(cin>>c){
if(c==1)cin>>x,q.push(x);
else if(c==2){if(q.empty())cout<<"UF";else cout<<q.front(),q.pop();}
else if(c==3){auto t=q;while(!t.empty())cout<<t.front()<<" ",t.pop();}
else break;}}
```

2 Array List

```
#include <bits/stdc++.h>
using namespace std;
int main(){vector<int> a;int c,x,p;
while(cin>>c){if(c==1)cin>>x>>p,a.insert(a.begin()+p,x);
else if(c==2)cin>>p,a.erase(a.begin()+p);
else if(c==3)for(int i:a)cout<<i<<" ";
else break;}}
```

3A Linked Stack

```
#include <bits/stdc++.h>
using namespace std;
struct N{int d;N*n;N(int x):d(x),n(NULL){}};
int main(){N*t=NULL;int c,x;while(cin>>c){
if(c==1){cin>>x;N*r=new N(x);r->n=t;t=r;}
else if(c==2){if(!t)cout<<"UF";else cout<<t->d,t=t->n;}
else if(c==3)for(N*r=t;r;r=r->n)cout<<r->d<<" ";
else break;}}
```

3B Linked Queue

```
#include <bits/stdc++.h>
using namespace std;
struct N{int d;N*n;N(int x):d(x),n(NULL){}};
int main(){N*f=NULL,*r=NULL;int c,x;while(cin>>c){
if(c==1){cin>>x;N*t=new N(x);if(!f)f=r=t;else r->n=t,r=t;}
else if(c==2){if(!f)cout<<"UF";else cout<<f->d,f=f->n;if(!f)r=NULL;}
else if(c==3)for(N*t=f;t;t=t->n)cout<<t->d<<" ";
else break;}}
```

4 Doubly Linked List

```
#include <bits/stdc++.h>
using namespace std;
struct D{int d;D*p,*n;D(int x):d(x),p(NULL),n(NULL){}};
int main(){D*h=NULL;int c,x;while(cin>>c){
if(c==1){cin>>x;D*t=new D(x);t->n=h;if(h)h->p=t,h=t;}
else if(c==2){cin>>x;D*t=h;while(t&& t->d!=x)t=t->n;
if(t){if(t->p)t->p->n=t->n;else h=t->n;if(t->n)t->n->p=t->p;}}
else if(c==3)for(D*t=h;t;t=t->n)cout<<t->d<<" ";
else break;}}
```

5 Polynomial

```
#include <bits/stdc++.h>
using namespace std;
struct T{int c,p;T*n;T(int a,int b):c(a),p(b),n(NULL){}};
T* ins(T*h,int c,int p){T*t=new T(c,p);
if(!h||p>h->p)t->n=h,h=t;
else{T*s=h;while(s->n&& s->n->p>=p)s=s->n;t->n=s->n;s->n=t;}return h;}
T* add(T*a,T*b){T*r=NULL;while(a||b){
if(a&&(!b||a->p>b->p))r=ins(r,a->c,a->p),a=a->n;
else if(b&&(!a||b->p>a->p))r=ins(r,b->c,b->p),b=b->n;
else r=ins(r,a->c+b->c,a->p),a=a->n,b=b->n;}return r;}
int main(){T*a=NULL,*b=NULL;int n,c,p;cin>>n;while(n--){cin>>c>>p,a=ins(a,c,p);
cin>>n;while(n--){cin>>c>>p,b=ins(b,c,p);T*r=add(a,b);
for(T*t=r;t;t=t->n)cout<<t->c<<"x^"<<t->p<<" ";}}
```

6 Infix to Postfix

```
#include <bits/stdc++.h>
using namespace std;
int pr(char c){return c=='+'||c=='-'?1:c=='*'||c=='/'?2:0;}
```

```
int main(){string s,o="";cin>>s;stack<char>st;
for(char c:s){if(isalnum(c))o+=c;
else if(c=='(')st.push(c);
else if(c==')'){while(st.top()!='(')o+=st.top(),st.pop();st.pop();}
else{while(!st.empty()&&pr(st.top())>=pr(c))o+=st.top(),st.pop();st.push(c);}}
while(!st.empty())o+=st.top(),st.pop();cout<<o;}
```

7A Linear Search

```
#include <bits/stdc++.h>
using namespace std;
int main(){int n,k;cin>>n;vector<int>a(n);
for(int&i:a)cin>>i;cin>>k;
for(int i=0;i<n;i++)if(a[i]==k){cout<<i;return 0;}
cout<<"NF";}
```

7B Binary Search

```
#include <bits/stdc++.h>
using namespace std;
int main(){int n,k;cin>>n;vector<int>a(n);
for(int&i:a)cin>>i;cin>>k;int l=0,h=n-1;
while(l<=h){int m=(l+h)/2;
if(a[m]==k){cout<<m;return 0;}
if(a[m]<k)l=m+1;else h=m-1;}cout<<"NF";}
```

8 AVL Tree

```
#include <bits/stdc++.h>
using namespace std;
struct N{int d,h;N*l,*r;N(int x):d(x),h(1),l(NULL),r(NULL){}};
int H(N*n){return n?n->h:0;}
int bf(N*n){return H(n->l)-H(n->r);}
N* rr(N*y){N*x=y->l;N*T=x->r;x->r=y;y->l=T;
y->h=max(H(y->l),H(y->r))+1;x->h=max(H(x->l),H(x->r))+1;return x;}
N* rl(N*x){N*y=x->r;N*T=y->l;y->l=x;x->r=T;
x->h=max(H(x->l),H(x->r))+1;y->h=max(H(y->l),H(y->r))+1;return y;}
N* ins(N*n,int k){if(!n)return new N(k);
if(k<n->d)n->l=ins(n->l,k);else n->r=ins(n->r,k);
n->h=max(H(n->l),H(n->r))+1;int b=bf(n);
if(b>1&&k<n->l->d)return rr(n);
if(b<-1&&k>n->r->d)return rl(n);
if(b>1&&k>n->l->d){n->l=rl(n->l);return rr(n);}
if(b<-1&&k>n->r->d){n->r=rr(n->r);return rl(n);}
return n;}
```

```

if(b<-1&&k<n->r->d){n->r=rr(n->r);return rl(n);}return n;}
void in(N*n){if(!n)return;in(n->l);cout<<n->d<<" ";in(n->r);}
int main(){N*r=NULL;int x;while(cin>>x)r=ins(r,x);in(r);}

```

9A DFS

```

#include <bits/stdc++.h>
using namespace std;
int g[20][20],vis[20],n;
void dfs(int v){vis[v]=1;cout<<v<<" ";for(int i=0;i<n;i++)
if(g[v][i]&&!vis[i])dfs(i);}
int main(){int e,a,b;cin>>n>>e;
while(e--){cin>>a>>b,g[a][b]=g[b][a]=1;
cin>>a;dfs(a);}

```

9B BFS

```

#include <bits/stdc++.h>
using namespace std;
int g[20][20],vis[20],n;
int main(){int e,a,b;cin>>n>>e;
while(e--){cin>>a>>b,g[a][b]=g[b][a]=1;
cin>>a;queue<int>q;q.push(a);vis[a]=1;
while(!q.empty()){int v=q.front();q.pop();cout<<v<<" ";
for(int i=0;i<n;i++)if(g[v][i]&&!vis[i])vis[i]=1,q.push(i);} }

```

10A Insertion Sort

```

#include <bits/stdc++.h>
using namespace std;
int main(){int n;cin>>n;vector<int>a(n);
for(int&i:a)cin>>i;for(int i=1;i<n;i++){
int k=a[i],j=i-1;while(j>=0&& a[j]>k)a[j+1]=a[j],j--;
a[j+1]=k;}for(int i:a)cout<<i<<" ";}

```

10B Bubble Sort

```

#include <bits/stdc++.h>
using namespace std;
int main(){int n;cin>>n;vector<int>a(n);
for(int&i:a)cin>>i;for(int i=0;i<n;i++)
for(int j=0;j<n-i-1;j++)
if(a[j]>a[j+1])swap(a[j],a[j+1]);
for(int i:a)cout<<i<<" ";}

```

10C Merge Sort

```
#include <bits/stdc++.h>
using namespace std;
void mg(vector<int>&a,int l,int m,int h){
vector<int>t;int i=l,j=m+1;
while(i<=m&&j<=h)t.push_back(a[i]<a[j]?a[i++]:a[j++]);
while(i<=m)t.push_back(a[i++]);while(j<=h)t.push_back(a[j++]);
for(int k=l;k<=h;k++)a[k]=t[k-l];}
void ms(vector<int>&a,int l,int h){
if(l<h){int m=(l+h)/2;ms(a,l,m);ms(a,m+1,h);mg(a,l,m,h);}}
int main(){int n;cin>>n;vector<int>a(n);
for(int&i:a)cin>>i;ms(a,0,n-1);
for(int i:a)cout<<i<<" ";
```

10D Quick Sort

```
#include <bits/stdc++.h>
using namespace std;
int part(vector<int>&a,int l,int h){
int p=a[h],i=l-1;for(int j=l;j<h;j++)
if(a[j]<p)swap(a[++i],a[j]);
swap(a[i+1],a[h]);return i+1;}
void qs(vector<int>&a,int l,int h){
if(l<h){int p=part(a,l,h);
qs(a,l,p-1);qs(a,p+1,h);}}
int main(){int n;cin>>n;vector<int>a(n);
for(int&i:a)cin>>i;qs(a,0,n-1);
for(int i:a)cout<<i<<" ";
```

11 Heap Priority Queue

```
#include <bits/stdc++.h>
using namespace std;
int main(){priority_queue<int>h;int c,x;
while(cin>>c){if(c==1)cin>>x,h.push(x);
else if(c==2){if(h.empty())cout<<"UF";else cout<<h.top(),h.pop();}
else if(c==3){auto t=h;while(!t.empty())cout<<t.top()<<" ",t.pop();}
else break;}}
```

12 Hashing Linear Probing

```
#include <bits/stdc++.h>
using namespace std;
```

```
int main(){int n=10,h[10];fill(h,h+10,-1);
int c,x;while(cin>>c){
if(c==1){cin>>x;int i=x%n;while(h[i]!=-1)i=(i+1)%n;h[i]=x;}
else if(c==2){cin>>x;int i=x%n;while(h[i]==-1&&h[i]!=x)i=(i+1)%n;
cout<<(h[i]==x?"F":"NF");}
else if(c==3){for(int i=0;i<n;i++)cout<<h[i]<<" ";}
else break;}}
```